

APPLICATION UNDER UNITED STATES PATENT LAWS

Atty. Dkt. No. PW 0275015
(M#)

Invention: METHOD AND SYSTEM FOR MAPPING NETWORK ATTACKS ONTO A STRATEGY GAME

Inventor (s): LORTZ, Victor B.

Pillsbury Winthrop LLP
Intellectual Property Group
1100 New York Avenue, NW
Ninth Floor
Washington, DC 20005-3918
Attorneys
Telephone: (202) 861-3000

This is a:

- ☐ Provisional Application
- ☒ Regular Utility Application
- ☐ Continuing Application
 - ☐ The contents of the parent are incorporated by reference
- ☐ PCT National Phase Application
- ☐ Design Application
- ☐ Reissue Application
- ☐ Plant Application
- ☐ Substitute Specification
 - Sub. Spec Filed _____
 - in App. No. _____ / _____
- ☐ Marked up Specification re
 - Sub. Spec. filed _____
 - In App. No. _____ / _____

SPECIFICATION

METHOD AND SYSTEM FOR MAPPING NETWORK ATTACKS ONTO A STRATEGY GAME

5

BACKGROUND

1. Field

This invention relates in general to computer network security. Specifically, this invention relates to methods and systems for mapping network attacks.

2. General Background and Related Art

A goal of network security is to ensure safe and reliable network operations. Various network security components, such as firewalls, are designed to prevent illegal intrusions. These components often monitor network activities and record various types of information relating to these activities. For example, firewalls may keep a log file that lists all accesses to the computer systems protected by the firewalls. For each access, the ID of the user who made the access, the time of the access, the location of the access, and the manner in which the access took place (remotely or locally) may be recorded. Log files generated by network security components are typically voluminous.

To detect a security breach such as an attack by a hacker, one needs to understand the information embedded in log files. However, interpretation of log files, which contain information indicating the severity and extent of network problems, may require significant technical expertise. Attempts have been made to transform raw log data into understandable forms. For example, color-coded histograms may be generated from raw log files to provide graphical views of the information embedded therein. Yet, interpretation of histograms also requires substantial technical knowledge.

Log files generated by network security monitoring tools may contain vendor-specific types of information presented in distinct forms. This increases the burden on

network system administrators and other users. For example, the log file for a particular type of firewall, generated by vendor A's security monitoring system, may be very different from the log file for a server computer, generated by vendor B's security monitoring system. The two log files may include different types of information in
5 different vendor-specific formats.

The Intrusion Detection Message Exchange Format (IDMEF) has recently been proposed to define data formats and exchange procedures for sharing information between different network security tools. However, this format retains the low-level data semantics of log files generated by vendor-specific network security tools. Significant effort and
10 technical expertise are required to analyze and understand IDMEF log files.

Therefore, what is needed is a system and method to effectively convey different types of log data to users.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a high-level diagram illustrating a principle of the present invention.

FIG. 2 is a block diagram of a system according to an embodiment of the present invention.

FIG. 3 is a block diagram of a transformer according to an embodiment of the
20 present invention.

FIG. 4 is a flow diagram illustrating a method according to an embodiment of the present invention.

DETAILED DESCRIPTION

The following detailed description refers to the accompanying drawings that illustrate exemplary embodiments of the present inventions. Other embodiments are possible and modifications may be made to the embodiments without departing from the spirit and scope of the invention. Therefore, the following detailed description is not meant to limit the invention. Rather, the scope of the invention is defined by the appended claims.

It will be apparent to one of ordinary skill in the art that the embodiments as described below may be implemented in many different embodiments of software, firmware, and hardware in the entities illustrated in the figures. The actual software code or specialized control hardware used to implement the present invention is not limiting of the present invention. Thus, the operation and behavior of the embodiments will be described without specific reference to the actual software code or specialized hardware components. The absence of such specific references is feasible because it is clearly understood that artisans of ordinary skill would be able to design software and control hardware to implement the embodiments of the present invention based on the description herein with only a reasonable effort and without undue experimentation.

Moreover, the processes associated with the presented embodiments may be stored in any storage device, such as, for example, a computer system (non-volatile) memory, an optical disk, magnetic tape, or magnetic disk. Furthermore, the processes may be programmed when the computer system is manufactured or via a computer-readable medium at a later date. Such a medium may include any of the forms listed above with respect to storage devices and may further include, for example, a carrier wave modulated, or otherwise manipulated, to convey instructions that can be read, demodulated/decoded and executed by a computer.

A method and system for mapping network attacks onto a strategy game, as described herein, involves receiving at least one log file generated by network security monitoring tools. The log file is received by a transformer. The transformer transforms the log file into a set of characters with associated action inputs, such that each of the
5 characters is associated with an action input.

Some aspects of network security relate to aspects of computer games. For instance, network attacks, responses, resource acquisition, vulnerabilities, territory, and the like may have analogues in computer games. As such, network events contained in security log files may be represented by various characters taking various actions within a
10 strategy game. Accordingly, a network administrator may view security data as if it were a recorded session of a computer game, such as Microsoft Corporation's Age of Empires. For instance, a port scan initiated by a remote hacker may appear as a scout character moving along a fortified wall searching for vulnerabilities. If a vulnerable port is found by the hacker, a subsequent attack on the port by the hacker may be represented by a directed
15 attack by another character from the same army as the scout. Such a mapping may convey network status in an intuitive, comprehensive manner.

FIG. 1 is a high-level diagram illustrating a principle of the present invention. As shown, one or more log files 101 are transformed into a representation, such as a strategy game 110. Strategy game 110 may be shown on a display of a computer 120. Log files
20 101 may comprise various vendor-specific files generated by off-the-shelf network security monitoring tools. Log files 101 may also comprise shareware or proprietary network security monitoring tools. Such files may chart network activity from various network vantage points.

FIG. 2 is a high-level block diagram of system 200 according to an embodiment of
25 the present invention. System 200 comprises transformer 210, strategy game renderer

230, and display unit 250. Transformer 210 receives one or more log files 201 generated by network security monitoring tools. Transformer 210 may process different types of log files 201. Transformer 210 transforms log files 201 into a set of characters with associated action inputs. Each of the characters may be associated with an action input. The set of
5 characters with associated action inputs may comprise a game session 220.

Game session 220 may comprise a file saved in the format of a game session of a strategy game. Strategy games may have vendor-specific game session formats. Thus, transformer 230 may output game sessions 220 of differing formats.

Strategy game renderer 230 may comprise strategy game software. Strategy game
10 renderer 230 receives game session 220. If game session 220 is compatible with strategy game renderer 230, strategy game renderer 230 renders game session 220 in the form of a strategy game 240, such as Age of Empires. Strategy game 240 may be displayed on display unit 250.

FIG. 3 is a high-level block diagram illustrating transformer 300 according to an
15 embodiment of the present invention. Transformer 300 comprises receiver 310 and transformer agent 320.

Receiver 310 receives one or more log files 301 generated by network security monitoring tools. Transformer agent 320 may convert log files 301 into zero or more event representations. Such event representations may be vendor-independent. For
20 example, transformer agent 320 may parse a log file 301, analyze log file 301, and identify that a port scan has been undertaken by a remote hacker. This event may be represented by an event representation, which may include a unique code. Each identified port scan may be designated by the same code. Transformer agent 320 may then map codes designating port scans and other such event representations to one or more game sessions

25 360.

Transformer agent 320 may comprise a number of subsidiary agents, such as one or more log file transformer agents 330 and one or more security event transformer agents 350. Log file transformer agents 330 may convert log files 301 into zero or more event representations 340. Security event transformer agents 350 may map the event representations 340 into a set of characters with associated action inputs, such as game session 360. Although FIG. 3 shows one log file transformer agent 330 and one security event transformer agent 350, multiple agents may be included in other embodiments of transformer 300.

Each type of log file 301 may have a corresponding log file transformer agent 330, which converts that type of log file into one or more vendor-independent event representations 340. In other words, each network security monitoring tool that generates a type of log file may have a corresponding log file transformer agent 330. Conversion of log file 301 into event representations 340 may comprise a number of steps. In some embodiments, log file transformer agent 330 may analyze each of the log files 301 with which log file transformer agent 330 is compatible, detect zero or more events based on the results of the analysis, and generate event representations 340 for the events detected. In an exemplary implementation, event representations may be arranged in a canonical form, such as one embodied in an XML file.

Security event transformer agent 350 may map event representations 340 into a vendor-specific game session 360 by various techniques. For instance, event representations 340 may be processed, a character may be selected or assigned to each of the event representations 340, and an action input to associate with each of the selected characters may be determined based on each of the event representations 340. The set of characters and associated action inputs may then be stored as game session 240.

Each type of security event transformer agent 350 may correspond to a particular strategy game. For instance, a security event transformer agent 350 for Age of Empires may produce an Age of Empires game session. In embodiments that include multiple security event transformer agents 350, multiple game sessions may be produced. As such,
5 the same set of log data may be viewed within the context of a number of strategy games.

FIG. 4 is a flow diagram of method 400 according to an embodiment of the present invention. In block B401, a log file is received. In block B410, the log file is converted into one or more event representations. These event representations are mapped to a set of characters and associated action inputs in block B420. The set of characters and
10 associated action inputs is then rendered as a strategy game in block B430. The results of the rendering may be displayed on a screen, as indicated by block B440.

The foregoing description of the preferred embodiments is provided to enable any person skilled in the art to make or use the present invention. Various modifications to these embodiments are possible, and the generic principles presented herein may be
15 applied to other embodiments as well. For instance, the present invention may be implemented in contexts other than network security contexts, such as in voicemail, e-mail, or other such messaging systems. Log files detailing messaging activities by various remote callers or senders may be mapped to a set of characters and associated action inputs, and rendered as a strategy game.

Moreover, the invention may be implemented in part or in whole as a hard-wired circuit, as a circuit configuration fabricated into an application-specific integrated circuit, or as a firmware program loaded into non-volatile storage or a software program loaded from or into a data storage medium as machine-readable code, such code being
20 instructions executable by an array of logic elements such as a microprocessor or other
25 digital signal processing unit.

As such, the present invention is not intended to be limited to the embodiments shown above but rather is to be accorded the widest scope consistent with the principles and novel features disclosed in any fashion herein.

Intel Ref: P10428
Pillsbury Ref: GJP/81674/275015/CMC